

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
INFORMÁTICA
GRADO EN INGENIERÍA DEL SOFTWARE

**ENTORNO DE INFORMACIÓN Y GESTIÓN PARA EL
ALUMNADO DE LA ETSII
INFORMATION ENVIRONMENT AND MANAGEMENT FOR
THE ETSII STUDENT BODY**

Realizado por
Enrique Ladrón de Guevara Santaella
Tutorizado por
Eduardo Guzmán De los Riscos
Departamento
Lenguajes y Ciencias de la Computación

UNIVERSIDAD DE MÁLAGA
MÁLAGA, 12/2015

Fecha defensa:
El Secretario del Tribunal

Resumen y palabras clave

El “Entorno de información y gestión para el alumnado de la ETSII” es una herramienta que permite la consulta de los datos más relevantes en el día a día de un alumno y realizar tareas simples pero de uso frecuente.

El sistema se compone de dos partes, una aplicación web y una aplicación móvil para dispositivos Android. La aplicación web se encarga de leer los datos de entrada a partir de distintos archivos con extensión .csv y los almacena en una base de datos MySQL, mientras que la aplicación móvil recibe esos datos mediante servicios REST y ejecuta toda la funcionalidad con los mismos.

El conjunto de funcionalidades que ofrece la aplicación requieren la autenticación en la misma a través de una cuenta de usuario en Google. El sistema además permite la generación de los horarios del alumno, adición de los exámenes al calendario personal del usuario, consulta de datos del profesorado, concertado de tutorías, asistente de matriculación y consulta del calendario lectivo de la titulación correspondiente.

Con todo ello se pretende agilizar la consulta de datos relevantes de uso frecuente por parte del alumnado de la E.T.S. de Ingeniería Informática en la web del centro, el campus virtual y la DUMA.

Palabras clave: Alumnado, ETSII, consulta, datos relevantes, tareas, frecuentes.

Abstract and keywords

The “Information environment and management for the ETSII student body” is a tool that allows the enquiries of the student’s day-to-day most relevant information and to carry out simple but frequent tasks.

The system consists of two parts, a web app and a mobile app for Android devices. The web app handles the data imported from different .csv files and stores them in a MySQL database. Then, the mobile app gathers all those data through REST services and performs the whole functionality.

This functionality consists of a login mechanism through a Google user account, the creation of the student’s timetable, the addition of the exams to the user’s personal calendar, search of the teachers’ information, arrangement of tutorial hours, registration wizard, and checks to the scholar calendar of the corresponding degree.

All this is expected to speed up and ease the student body’s frequent information queries through the school website, the Virtual Campus and the DUMA.

Keywords: student body, ETSII, query, relevant information, tasks, frequent.

ÍNDICE

CAPÍTULO 1. Introducción	1
1.1 Contexto	1
1.2 Motivaciones	1
1.3 Descripción General	2
1.4 Fases del proyecto	3
1.4.1 Fase de Análisis	3
1.4.2 Fase de Diseño	4
1.4.3 Fase de Desarrollo	4
1.5 Tecnologías utilizadas	5
1.5.1 Android	5
1.5.2 CSV	6
1.5.3 REST	6
1.5.4 JSON	7
 CAPÍTULO 2. Especificación	 9
2.1 Catálogo de Objetivos	9
2.2 Catálogo de Requisitos	11
2.2.1 Requisitos Funcionales	11
2.2.2 Requisitos No Funcionales	16
2.2.3 Requisitos de Información	17
2.3 Base de Datos	18
2.4 Catálogo de Normas	19
2.5 Modelo de Casos de Uso de Análisis	20
 CAPÍTULO 3. Conclusiones y trabajos futuros	 28
3.1 Trabajos futuros	28
3.2 Conclusiones	29
 Referencias bibliográficas	 32
 APÉNDICE A. Términos y definiciones usados en la memoria	 33

CAPÍTULO 1. INTRODUCCIÓN

1.1 Contexto

En la actualidad, la Escuela Técnica Superior de Ingeniería Informática posee una gran infraestructura en materia de tecnologías de la información. Dispone tanto del campus virtual, la web propia de la escuela como de la DUMA.

A pesar de todo ello, los alumnos no tienen una forma rápida y cómoda de consultar sus datos cotidianos en el devenir del curso académico. Además esta información se encuentra dispersa entre los distintos portales y no existe una forma conjunta de consultar esos datos. Asimismo las webs mencionadas anteriormente no disponen de una versión móvil, dejando en evidencia la poca comodidad que experimentan los usuarios al navegar por ellas desde sus dispositivos móviles.

Teniendo en cuenta que los dispositivos móviles tienen una cuota de mercado cada vez mayor, se hace aún más necesario el acercar el contenido de éstas a dichos dispositivos.

1.2 Motivaciones

Durante mi experiencia personal a lo largo del estudio de la titulación “Grado en Ingeniería del Software” se puso de manifiesto la cantidad de datos que consulta el alumnado en su día a día en lo referente a su expediente académico y relacionados. Y esa ingente cantidad de consultas se realizaban a través de los medios disponibles, los cuales eran muy poco atractivos para el usuario.

Sufriendo este problema durante los numerosos años de la titulación, cuando llegó el momento de elegir una línea de proyecto para el trabajo de fin de grado, se dieron dos circunstancias diferentes pero relacionadas entre sí.

No encontré una línea de proyecto que me atrajese lo suficiente como para estar trabajando meses en ella. Ya sea por la rama en la que se fundamentaban o bien por los aspectos técnicos que requerían.

Por otro lado, quería subsanar lo que para mí era un defecto muy importante para el alumnado; crear una herramienta que solucionase dicho problema y que fuera algo realmente con una utilidad práctica.

Dadas las circunstancias, lo mejor era proponer una línea nueva, con la intención de solucionar precisamente los detalles que yo pensaba que eran una necesidad para el resto de compañeros.

1.3 Descripción General

Con la idea de facilitar el acceso de los alumnos a la información de estos portales de una manera cómoda y unificada se pensó una aplicación móvil con la cual el alumno pudiese consultar los datos, que con mayor frecuencia utiliza de los portales que ofrece la ETSII, de una manera conjunta, más rápida y eficiente.

Las razones para realizar una aplicación móvil en lugar de hacer una adaptación móvil de las webs ya existentes fueron las siguientes: La dispersión de los datos en diferentes portales y la posibilidad de usar ciertos servicios de la aplicación sin necesidad de una conexión a internet.

A la hora de elegir el sistema operativo para el que se desarrollaría dicha aplicación se llegó a la conclusión de que la mejor opción era el sistema operativo desarrollado por Google: Android. Esto se debe al altísimo porcentaje de usuarios que navegan por la web a través de dispositivos que usan este sistema operativo respecto a los demás. Aunque la opción de Apple también se tuvo en cuenta debido al gran éxito que están experimentando los dispositivos de dicha marca se desestimó porque el número de dispositivos con Android seguía siendo abrumadoramente mayor y la tendencia no indicaba otra cosa que no fuera la continua expansión de los mismos.

La aplicación se pensó en dos partes. La primera constaría de una aplicación web montada sobre un servidor “GlassFish”, que estaría conectada a la base de datos que a su vez usaría MySQL. Esta aplicación exportaría las entidades de la base de datos a través de una serie de servicios REST.

La segunda parte del proyecto consistiría en la propia aplicación para Android. Esta aplicación al estar ejecutándose sobre el sistema operativo Android usa aplicaciones y datos del propio dispositivo, tales como la cuenta de Google del usuario, las aplicaciones de Calendar y Gmail que vienen de fábrica en el terminal. La aplicación además consumirá los servicios REST generados anteriormente en la aplicación web para hacer una carga inicial de datos con los que trabajar más adelante.

El usuario una vez que se ha autenticado en la aplicación tendrá a su disposición una serie de funcionalidades que le ayudarán a lo largo del curso. Tales como:

- Consultar el horario del alumno.
- Consultar el calendario lectivo.
- Sincronizar los exámenes finales que tenga el alumno con su aplicación de Google Calendar.
- Consultar los datos de un profesor tales como despacho, número de teléfono de su despacho, horario de tutorías, nombre y apellidos, etc.
- Solicitar una tutoría con alguno de los profesores de alguna de las asignaturas en las que esté matriculado el alumno.
- Utilizar un asistente de matriculación que ayudará al alumno a la hora de elegir sus asignaturas para el próximo año, calculando tanto la suma total de créditos matriculados como el montante final a pagar por ellos.

1.4 Fases del Proyecto

1.4.1 Fase de Análisis

Como ya hemos mencionado, existía una necesidad entre el alumnado de la Escuela Técnica Superior de Ingeniería Informática de poder consultar sus respectivos datos sin necesidad de tener que buscar en cualquiera de los tres portales de los que disponía la entidad.

Esto se daba porque dichos portales no estaban adaptados a los dispositivos móviles, los cuales son los que mayor uso tienen por parte de los alumnos mientras están en los descansos entre clases y circunstancias similares.

Además la información al estar disgregada por esos tres lugares resultaba realmente poco práctico y muy engorroso. Todo eso sumado a que algunos dispositivos con algunas compañías no tienen cobertura en la totalidad de las instalaciones de la escuela, era necesario que la mayoría de las funcionalidades pudieran operar sin conexión.

Debido a que los datos relevantes a consultar por el alumno son datos particulares y personales, era necesario dividir el proyecto en dos partes. Una aplicación web que contuviese toda la información en su base de datos. Y una aplicación móvil que recibiese de la aplicación web solo los datos que le correspondan, dejando así al usuario solo con los datos que le pertenecen relacionados con su matrícula y no los de sus respectivos compañeros.

Con todo esto planteado, se prosiguió con la elaboración del catálogo de requisitos, actores y catálogo de normas.

1.4.2 Fase de Diseño

A continuación se llevó a cabo la elaboración de los documentos que son consecuencia de los productos anteriormente obtenidos en la fase de análisis, esto es, la elaboración de los casos de uso, mockups, etc.

En esta parte del proyecto se utilizaron herramientas como Google Drive para la elaboración de los documentos, Trello para la organización de las tareas a realizar (este a su vez se seguiría utilizando en el resto de las fases del proyecto) y Pencil para confeccionar los mockups que más adelante sirvieron para acelerar el proceso de desarrollo del proyecto (no incluidos en este documento).

En todo momento se realizaron consultas cautelares al tutor del TFG para el correcto asesoramiento en una fase tan crucial como la fase de diseño para aclarar las dudas y pequeñas correcciones que de haberse descubierto en las siguientes fases del proyecto hubieran causado grandes retrasos e inconvenientes.

1.4.3 Fase de Desarrollo

Durante la fase de desarrollo en los meses siguientes se utilizó la plataforma web de gestión de proyecto y repositorios GitHub. Esto permitió la posibilidad de trabajar en distintas máquinas, un correcto control de versiones y un sistema de backups cuya utilización no fue necesaria en toda la fase de desarrollo.

La aplicación web se desarrolló utilizando el IDE NetBeans y el servidor GlassFish. Se utilizó una base de datos MySQL mientras que para el desarrollo de la aplicación móvil se usó Android Studio por ser más ligero y tener una mayor usabilidad que otras alternativas como por ejemplo Eclipse.

1.5 Tecnologías utilizadas

1.5.1 Android

Inicialmente fue desarrollado por Android Inc., empresa que Google respaldó económicamente y más tarde, en 2005, compró.

La plataforma de hardware principal de Android es la arquitectura ARM. Hay soporte para x86 en el proyecto Android-x86, y Google TV utiliza una versión especial de Android x86. Sus aplicaciones están escritas en lenguaje de programación Java.

Los desarrolladores tienen acceso completo a los mismos APIs del framework usados por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de componentes; cualquier aplicación puede publicar sus capacidades y cualquier otra aplicación puede luego hacer uso de esas capacidades (sujeto a reglas de seguridad del framework). Este mismo mecanismo permite que los componentes sean reemplazados por el usuario.

Android incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android; algunas son: System C library (implementación biblioteca C estándar), bibliotecas de medios, bibliotecas de gráficos, 3D y SQLite, entre otras.

Android incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Dalvik ha sido escrito de forma que un dispositivo puede correr múltiples máquinas virtuales de forma eficiente. Dalvik ejecuta archivos en el formato Dalvik Executable (.dex), el cual está optimizado para memoria mínima. La Máquina Virtual está basada en registros y corre clases compiladas por el compilador de Java que han sido transformadas al formato.dex por la herramienta incluida "dx".

Android depende de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software.

1.5.2 CSV

Los archivos CSV (del inglés comma-separated values) son un tipo de documento en formato abierto sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas (o punto y coma en donde la coma es el separador decimal) y las filas por saltos de línea. Los campos que contengan una coma, un salto de línea o una comilla doble deben ser encerrados entre comillas dobles.

El formato CSV es muy sencillo y no indica un juego de caracteres concreto, ni cómo van situados los bytes, ni el formato para el salto de línea. Estos puntos deben indicarse al abrir el archivo.

1.5.3 REST

La Transferencia de Estado Representacional (Representational State Transfer) o REST es un estilo de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web.

Si bien el término REST se refería originalmente a un conjunto de principios de arquitectura, en la actualidad se usa en el sentido más amplio para describir cualquier interfaz entre sistemas que utilice directamente HTTP para obtener datos o indicar la ejecución de operaciones sobre los datos, en cualquier formato (XML, JSON, etc) sin las abstracciones adicionales de los protocolos basados en patrones de intercambio de mensajes, como por ejemplo SOAP. Los sistemas que siguen los principios REST se llaman con frecuencia RESTful.

REST afirma que la web ha disfrutado de escalabilidad como resultado de una serie de diseños fundamentales clave:

Un protocolo cliente/servidor sin estado: cada mensaje HTTP contiene toda la información necesaria para comprender la petición. Como resultado, ni el cliente ni el servidor necesitan recordar ningún estado de las comunicaciones

entre mensajes. Sin embargo, en la práctica, muchas aplicaciones basadas en HTTP utilizan cookies y otros mecanismos para mantener el estado de la sesión.

Un conjunto de operaciones bien definidas que se aplican a todos los recursos de información: HTTP en sí define un conjunto pequeño de operaciones, las más importantes son POST, GET, PUT y DELETE. Con frecuencia estas operaciones se equiparan a las operaciones CRUD en bases de datos que se requieren para la persistencia de datos, aunque POST no encaja exactamente en este esquema.

Una sintaxis universal para identificar los recursos. En un sistema REST, cada recurso es direccionable únicamente a través de su URI.

El uso de hipermedios, tanto para la información de la aplicación como para las transiciones de estado de la aplicación: la representación de este estado en un sistema REST son típicamente HTML o XML. Como resultado de esto, es posible navegar de un recurso REST a muchos otros, simplemente siguiendo enlaces sin requerir el uso de registros u otra infraestructura adicional.

1.5.4 JSON

JSON, acrónimo de JavaScript Object Notation, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

La simplicidad de JSON ha dado lugar a la generalización de su uso, especialmente como alternativa a XML en AJAX. Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es mucho más sencillo escribir un analizador sintáctico (parser) de JSON. En JavaScript, un texto JSON se puede analizar fácilmente usando la función “eval()”, lo cual ha sido fundamental para que JSON haya sido aceptado por parte de la comunidad de desarrolladores AJAX, debido a la ubicuidad de JavaScript en casi cualquier navegador web.

En la práctica, los argumentos a favor de la facilidad de desarrollo de analizadores o del rendimiento de los mismos son poco relevantes, debido a las cuestiones de seguridad que plantea el uso de eval() y el auge del procesamiento nativo de XML incorporado en los navegadores modernos. Por esa razón, JSON se emplea habitualmente en entornos donde el tamaño del

flujo de datos entre cliente y servidor es de vital importancia cuando la fuente de datos es explícitamente de fiar y donde no es importante el no disponer de procesamiento XSLT para manipular los datos en el cliente.

Si bien es frecuente ver JSON posicionado contra XML, también es frecuente el uso de JSON y XML en la misma aplicación.

Cada vez hay más soporte de JSON mediante el uso de paquetes escritos por terceras partes. La lista de lenguajes soportados incluye ActionScript, C, C++, C#, ColdFusion, Common Lisp, Delphi, E, Eiffel, Java, JavaScript, ML, Objective-C, Objective CAML, Perl, PHP, Python, Rebol, Ruby, Lua y Visual FoxPro.

El término JSON está altamente difundido en los medios de programación, sin embargo, es un término mal descrito ya que en realidad es solo una parte de la definición del estándar ECMA-262 en que está basado Javascript.

CAPÍTULO 2. ESPECIFICACIÓN

2.1 Catálogo de Objetivos

A continuación se describirán todos los objetivos planteados para satisfacer las necesidades detectadas durante la fase de análisis. Cada uno de estos objetivos lleva un código único y una prioridad asociada a la relevancia de dicho objetivo:

Entrada de datos

(OB-01) - Prioridad Alta

La inyección de los datos en la base de datos se realizará a través de la lectura de los diferentes archivos csv por parte del servidor. A su vez estos datos serán exportados a la aplicación Android por medios de servicios.

Login con cuenta de Google

(OB-02) - Prioridad Media

Se pretende usar la autenticación de Google para así facilitar datos de interés a la propia aplicación sin que el usuario tenga que introducirlos manualmente ganando así en tiempo y en comodidad por parte del propio usuario. Además como está desarrollada en el sistema operativo Android no existe la posibilidad de que el usuario no posea una cuenta en dicho sistema.

Visualización del horario

(OB-03) - Prioridad Alta

Una de los objetivos más importantes, pues es de las primeras necesidades a cubrir. La aplicación visualiza el horario de las asignaturas en las que el alumno se encuentra matriculado.

Sincronizar con Google Calendar

(OB-04) - Prioridad Media

La aplicación añadirá al calendario personal del alumno en Google Calendar un evento por cada examen de las asignaturas de las que está matriculado con todos los datos relacionado con cada examen, tales como aula, fecha y hora.

Concertar tutorías

(OB-05) - Prioridad Alta

Se pretende poder concertar una tutoría con cualquiera de los profesores que impartan al menos una de las asignaturas de las que el usuario está matriculado. Se realizará mediante el correo electrónico.

Asistente de matriculación

(OB-06) - Prioridad Alta

Un asistente que permita la visualización de una supuesta matrícula. En definitiva se trata de que el alumno seleccione las asignaturas a visualizar, y se genere un horario automáticamente con dichas asignaturas para poder apreciar de manera rápida y visual si hay coincidencias de horarios y cualquier otra incidencia. También se realizará un cálculo tanto de los créditos a matricular como del coste económico total de la misma.

Visualización del calendario lectivo

(OB-07) - Prioridad Baja

Una visualización a pantalla completa del calendario lectivo del año en el que el alumno se encuentra matriculado.

2.2 Catálogo de Requisitos

En este apartado se detallarán todos los requisitos de nuestro sistema. El conjunto de todos ellos deberán cumplir todos los objetivos marcados anteriormente.

Los requisitos se dividen en funcionales, no funcionales y de información. Los requisitos funcionales son acciones concretas que deberá realizar la aplicación. Los no funcionales sin embargo son restricciones que debe cumplir el propio sistema. A su vez los requisitos de información son aquellas entidades que de una manera u otra el sistema debe almacenar. Generalmente esto da lugar posteriormente a la especificación de la base de datos del sistema.

2.2.1 Requisitos Funcionales

Cada requisito consta de:

- Nombre: Nombre del mismo.
- Código: Identificador único del requisito.
- Prioridad: Relevancia dentro del sistema.
- Descripción: Breve explicación sobre en qué consiste el requisito en cuestión.

Nombre	Lectura de datos de alumnos
Código	RF-01
Prioridad	Alta
Descripción	Lectura del archivo csv correspondiente a la ruta indicada y llenado de las entidades de la base de datos correspondientes.

Nombre	Lectura de datos de asignaturas
Código	RF-02
Prioridad	Alta
Descripción	Lectura del archivo csv correspondiente a la ruta indicada y llenado de las entidades de la base de datos correspondientes.

Nombre	Lectura de datos de profesores
Código	RF-03
Prioridad	Alta
Descripción	Lectura del archivo csv correspondiente a la ruta indicada y llenado de las entidades de la base de datos correspondientes.

Nombre	Lectura de datos de horas de clase
Código	RF-04
Prioridad	Alta
Descripción	Lectura del archivo csv correspondiente a la ruta indicada y llenado de las entidades de la base de datos correspondientes.

Nombre	Login
Código	RF-05
Prioridad	Alta
Descripción	Acceso a la aplicación mediante el correo electrónico vinculado a una cuenta de Google de las sincronizadas en el dispositivo donde se está ejecutando la aplicación.

Nombre	Logout
Código	RF-06
Prioridad	Media
Descripción	Salida de la aplicación tras la cual el sistema se vacía de información a la espera de un nuevo login.

Nombre	Generación de horario
Código	RF-07
Prioridad	Alta
Descripción	Generación de un horario personal en función de las asignaturas matriculadas por el alumno en el presente curso.

Nombre	Sincronización con google calendar
Código	RF-08
Prioridad	Media
Descripción	Adición al calendario personal del usuario en Google Calendar de un evento por examen de cada asignatura en la cual esté matriculado el alumno con todos los datos del mismo.

Nombre	Datos de interés del profesorado
Código	RF-09
Prioridad	Baja
Descripción	Permite la visualización de los datos de interés de cualquier profesor que imparta al menos alguna de las asignaturas matriculadas por el usuario. Dichos datos son tales como nombre y apellidos, despacho, correo y horarios de tutoría.

Nombre	Concertar tutorías con profesores
Código	RF-10
Prioridad	Baja
Descripción	Permite seleccionar la asignatura y el profesor correspondiente y enviar un correo electrónico a través de la aplicación por defecto en el dispositivo rellenando automáticamente campos como el asunto y el destinatario.

Nombre	Visualización de asignaturas en el asistente de matriculación
Código	RF-11
Prioridad	Media
Descripción	Visualización gráfica del horario del alumno en función de las asignaturas seleccionadas manualmente en el filtrado.

Nombre	Añadir asignaturas en el asistente de matriculación
Código	RF-12
Prioridad	Media
Descripción	Añadir una asignatura al conjunto de las asignaturas visualizadas en el asistente.

Nombre	Quitar asignatura en el asistente de matriculación
Código	RF-13
Prioridad	Media
Descripción	Quitar una asignatura de la visualización del asistente.

Nombre	Coste económico de una matrícula
Código	RF-14
Prioridad	Baja
Descripción	Se calculará en Euros el montante total al que asciende la matriculación prevista por el usuario.

Nombre	Cálculo de créditos de una matrícula
Código	RF-15
Prioridad	Baja
Descripción	Cálculo total del número de créditos a matricular con la selección realizada por el usuario.

Nombre	Mostrar calendario lectivo
Código	RF-16
Prioridad	Baja
Descripción	Visualización del calendario lectivo para la titulación del alumno.

2.2.2 Requisitos No funcionales

Cada requisito consta de:

- Nombre: Nombre del mismo.
- Código: Identificador único del requisito.
- Prioridad: Relevancia dentro del sistema.
- Descripción: Breve explicación sobre en qué consiste el requisito en cuestión.

Nombre	Carga inicial de datos
Código	RNF-01
Prioridad	Alta
Descripción	Realización de la carga inicial de datos a través de los servicios REST solo de la información relacionada con el alumno que ha realizado previamente el login. Esta carga de datos debe hacerse de forma paralela a la ejecución del hilo principal de la aplicación para evitar posibles retardos en el mismo, que puedan causar una navegación poco fluida y frustrante.

Nombre	Visualización de asignaturas en el asistente de matriculación
Código	RNF-02
Prioridad	Media
Descripción	Si dos o más asignaturas seleccionadas en el asistente de matriculación coinciden en la misma hora y día, se mostrará pertinentemente al usuario de tal incidencia para su corrección.

2.2.3 Requisitos de Información

Cada requisito consta de:

- Nombre: Nombre del mismo.
- Código: Identificador único del requisito.
- Prioridad: Relevancia dentro del sistema.
- Descripción: Breve explicación sobre en qué consiste el requisito en cuestión.

Nombre	Alumno
Código	RI-01
Descripción	Se almacenarán todos los datos personales del alumno, estos son nombre, apellidos, expediente y correo personal.

Nombre	Asignatura
Código	RI-02
Descripción	Con las asignaturas se almacenarán sus fechas de exámenes, aula del examen, código, si es optativa o no, en qué semestre se cursa. Además se almacenará en qué aula se imparten las clases así como la hora de inicio y final de cada una de las mismas.

Nombre	Profesor
Código	RI-03
Descripción	Datos de contacto del profesor, así como su horario de tutorías, despacho, correo, nombre y apellidos.

Nombre	Titulación
Código	RI-04
Descripción	Titulaciones que se imparten en la ETSII.

2.3 Base de Datos

A continuación se detalla la base de datos generada a partir del conjunto de requisitos de información anteriormente identificados.

Las tablas creadas son las siguientes:

- **Alumno:** Almacena su número de expediente (clave primaria), nombre, apellidos y el correo dado en la matriculación del curso actual.
- **Titulación:** Contiene el nombre de cada titulación así como un identificador único (clave primaria).
- **Asignatura:** Almacena el nombre de la asignatura, la fecha de su examen en convocatoria ordinaria, la hora de dicho examen, el aula del mismo, si la asignatura es optativa o no, en qué semestre se imparte y el código de la asignatura (clave primaria).
- **Hora Asignatura:** Almacena un identificador único, el grupo en el que se está matriculado, día en el que se imparte esa clase, hora a la que empieza esa clase, hora de finalización de la misma, el aula donde se imparte y si es de obligatoria asistencia o no.
- **Profesor:** Almacena un identificador único, el nombre, los apellidos, el teléfono del despacho, su correo electrónico de la universidad, el número de su despacho y su correspondiente horario de tutorías.

A continuación se mostrará el esquema real de la base de datos, mostrando así las relaciones entre las tablas anteriormente mencionadas así como las nuevas tablas que surgen de las relaciones de muchos a muchos.

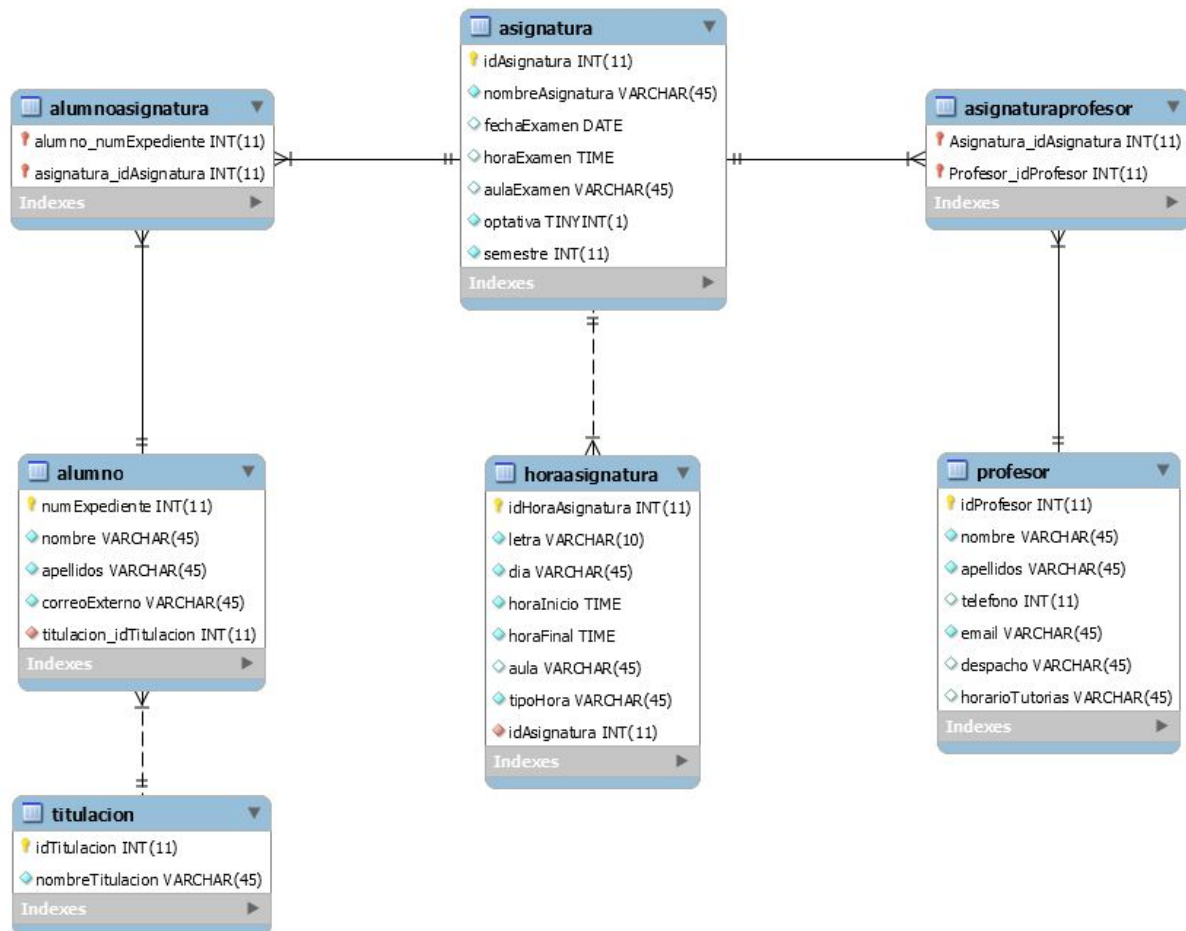


Figura 1. Esquema de la base de datos

2.4 Catálogo de Normas

El catálogo de normas del proyecto consiste en una serie de especificaciones o reglas que se deben cumplir en todo momento durante el desarrollo del mismo. Cada norma consta de:

- Nombre: Nombre de la misma.
- Código: Identificador único de la norma.
- Descripción: Breve explicación sobre en qué consiste la norma en cuestión.

Nombre	Estilo de escritura
Código	CN-01
Descripción	Para la fase de desarrollo, el nombre de variables y funciones se escribirá usando "UpperCamelCase".

Título	Colores corporativos
Código	CN-02
Descripción	Tema de colores de la Universidad de Málaga en toda la aplicación y su documentación.

2.5 Modelo de Casos de Uso de Análisis

Anteriormente hemos enumerado los requisitos funcionales que debe cumplir el proyecto, pero no especificamos en detalle el comportamiento de dichas funciones. Los casos de uso nos ayudarán a explicar estas funcionalidades y servirá de guía en la fase de desarrollo. Cada caso de uso contiene:

- Nombre: El nombre del caso de uso.
- Código: Identificador único.
- Descripción: Una breve descripción sobre en qué consiste el caso de uso.
- Escenario de éxito: La explicación paso por paso de al menos un escenario de éxito en la realización del caso de uso.
- Escenario de excepción: La explicación paso a paso de un escenario en el que no se consiga llevar a cabo el caso de uso en caso de que sea aplicable.
- Dependencias: Que casos de usos son necesarios que se cumplan anteriormente para poder ejecutar el mismo en cuestión.

Nombre	Lectura de datos
Código	CU-01
Descripción	Lectura de datos a partir de los archivos csv pertinentes por parte del servidor y volcado posterior en la base de datos.
Escenario de éxito	<ol style="list-style-type: none"> 1. Se arranca el servidor. 2. El servidor automáticamente llama a un método por cada entidad a leer, pasándole como parámetro de entrada la ruta del archivo csv correspondiente donde se encuentra el mismo. 3. Con los datos leídos el servidor llena la base de datos con ellos.
Escenario de excepción	<ol style="list-style-type: none"> 1. Se arranca el servidor. 2. El servidor automáticamente llama a un método por cada entidad a leer, pasándole como parámetro de entrada la ruta del archivo csv correspondiente donde se encuentra el mismo. 3. Los archivos no se encuentran en las rutas especificadas y no se pueden añadir nuevos datos a la base de datos.
Dependencias	N/A

Nombre	Login
Código	CU-01
Descripción	Acceso a la aplicación mediante el correo electrónico de su cuenta de Google mediante el cual se consumirán los servicios oportunos para la carga inicial de datos.
Escenario de éxito	<ol style="list-style-type: none"> 1. En la pantalla de login el usuario introduce correo electrónico. 2. Hace clic en el botón aceptar. 3. El sistema comprueba que existe dicho usuario. 4. El sistema consume el servicio REST correspondiente y efectúa la carga inicial de datos del sistema con sus datos. 5. Se redirige a la página principal de la app.

Escenario de excepción	<ol style="list-style-type: none"> 1. En la pantalla de login el usuario introduce su correo electrónico. 2. Hace clic en el botón aceptar. 3. El sistema comprueba que existe dicho usuario. 4. Al no existir ese número de expediente en la BD se informa al usuario mediante una alerta.
Dependencias	N/A

Nombre	Logout
Código	CU-02
Descripción	Salida de la aplicación tras la cual el sistema se vacía de información a la espera de un nuevo login.
Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario, desde la página principal, hace clic en el botón “Salir”. 2. La aplicación se vacía de toda la información cargada con la BD. 3. Se redirige al usuario a la página del login.
Escenario de excepción	N/A
Dependencias	Flujo: CU-01

Nombre	Generación de horario
Código	CU-03
Descripción	Generación de un horario personal en función de las asignaturas actualmente matriculadas por el usuario.

Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario accede a la vista principal. 2. Con los datos de la carga inicial de datos, se genera un horario en función a las asignaturas de la matrícula del usuario.
Escenario de excepción	<ol style="list-style-type: none"> 1. El usuario accede a la vista principal. 2. El usuario no tiene ninguna asignatura matriculada por lo que es imposible generar un horario. 3. Se informa al usuario del incidente mediante una alerta.
Dependencias	Flujo: CU-01

Nombre	Sincronización con google calendar
Código	CU-04
Descripción	Adición al calendario personal del usuario en la aplicación Google Calendar un evento por cada examen de las asignaturas matriculadas por el alumno.
Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario accede a la vista principal. 2. Hace clic en el botón “Sincronizar con Google Calendar”. 3. El sistema añade al calendario personal del usuario de su google calendar, añadiendo los eventos de los exámenes de sus respectivas asignaturas.
Escenario de excepción	<ol style="list-style-type: none"> 1. El usuario accede a la vista principal. 2. Hace clic en el botón “Sincronizar con Google Calendar”. 3. El usuario no tiene ninguna asignatura matriculada por lo que es imposible generar eventos con sus exámenes. 4. Se informa al usuario del incidente mediante una alerta.
Dependencias	Flujo: CU-1

Nombre	Consultar datos de los profesores
Código	CU-05
Descripción	Permite seleccionar la asignatura y el profesor del cual se quieren visualizar sus datos por pantalla.
Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario accede a la vista profesores. 2. Desde la vista de los profesores, el usuario selecciona la asignatura mediante un menú desplegable. 3. Selecciona el profesor de dicha asignatura mediante otro menú desplegable. 4. El sistema deja la aplicación en segundo plano y abre la aplicación de gmail precargando el email del destinatario con el del profesor seleccionado.
Escenario de excepción	N/A
Dependencias	Flujo: CU-01

Nombre	Concertar tutorías con profesores
Código	CU-06
Descripción	Permite seleccionar la asignatura y el profesor al cual se quiere enviar un mail a través de Gmail.
Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario accede a la vista profesores. 2. El usuario selecciona la asignatura de la cual desea una tutoría mediante un menú desplegable. 3. Selecciona el profesor de dicha asignatura con el cual quiere la tutoría mediante otro menú desplegable. 4. El usuario hace clic en "Concertar tutoría". 5. El sistema deja la aplicación en segundo plano y abre la aplicación de Gmail precargando el email del destinatario con el del profesor seleccionado.

Escenario de excepción	<ol style="list-style-type: none"> 1. El usuario accede a la vista profesores. 2. El usuario selecciona la asignatura de la cual desea una tutoría mediante un menú desplegable. 3. Selecciona el profesor de dicha asignatura con el cual quiere la tutoría mediante otro menú desplegable. 4. El usuario hace clic en “Concertar tutoría”. 5. El usuario no tiene la aplicación de gmail instalada en el dispositivo, por lo que se mostrará la versión web de gmail en el navegador del dispositivo.
Dependencias	Flujo: CU-01, CU-05

Nombre	Visualización de asignaturas en el asistente de matriculación
Código	CU-07
Descripción	Visualización gráfica del horario del usuario en función de las asignaturas seleccionadas por el usuario en el filtrado.
Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario accede a la vista del asistente. 2. Selecciona qué asignaturas quiere ver en la previsualización. 3. El sistema con esas asignaturas muestra sobre la vista el horario de las mismas.
Escenario de excepción	<ol style="list-style-type: none"> 1. El usuario accede a la vista del asistente. 2. Selecciona qué asignaturas quiere ver en la previsualización. 3. El sistema con esas asignaturas muestra sobre la vista el horario de las mismas. 4. Hay dos o más asignaturas que coinciden en hora y día, por lo que se avisa al usuario mediante una alerta.
Dependencias	Flujo: CU-01

Nombre	Adición de las asignaturas en el asistente de matriculación
Código	CU-08
Descripción	Añadir una asignatura al conjunto de las asignaturas visualizadas en el asistente.
Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario accede a la vista del asistente. 2. El usuario selecciona la asignatura de la columna correspondiente y hace clic en “Añadir”. 3. La asignatura desaparece de la columna “Todas las asignaturas”. 4. La asignatura aparece en la columna “Mis asignaturas” y se previsualiza en el horario.
Escenario de excepción	N/A
Dependencias	Flujo: CU-01, CU-07

Nombre	Quitar asignatura en el asistente de matriculación
Código	CU-08
Descripción	Quitar una asignatura de la visualización del asistente.
Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario accede a la vista del asistente. 2. El usuario selecciona la asignatura de la columna correspondiente y hace clic en “Quitar”. 3. La asignatura desaparece de la columna “Mis asignaturas”. 4. La asignatura aparece en la columna de “Todas las asignaturas” y desaparece del horario.
Escenario de excepción	N/A
Dependencias	Flujo: CU-01, CU-07, CU-08

Nombre	Mostrar calendario lectivo
Código	CU-09
Descripción	Visualización del calendario lectivo para la titulación del alumno.
Escenario de éxito	<ol style="list-style-type: none"> 1. El usuario hace clic en “Calendario lectivo” desde la vista principal. 2. Se carga la vista del calendario.
Escenario de excepción	N/A
Dependencias	Flujo: CU-01

CAPÍTULO 3. CONCLUSIONES Y TRABAJOS FUTUROS

3.1 Trabajos Futuros

Este proyecto, a pesar de todas las funciones implementadas, permite una gran cantidad de líneas de expansión. Existen una infinidad de funcionalidades que a pesar de ser altamente atractivas, por motivos de tiempo y adaptabilidad al número

de horas establecidas para un Proyecto de Fin de Grado, no se han implementado en este proyecto.

Pero me gustaría dejar indicadas al menos cuáles de esas supuestas funcionalidades consideraría interesante de implementar en un hipotético futuro sobre el proyecto actual.

- Sincronización de la aplicación móvil con el Campus Virtual, generando alertas en el móvil cuando corresponda con los eventos del propio campus virtual.
- Sincronización de la aplicación móvil con el Campus Virtual, añadiendo al Calendario de Google todos los eventos del propio campus virtual.
- Implantación de un sistema de comunicación entre usuarios de la misma titulación.
- Posibilidad de almacenar de alguna manera las hipotéticas matrículas que el usuario haga en el asistente de matriculación.

Esto a su vez podría permitir no solo guardarlas para su posterior visualización, sino poder exportarla a otros formatos para su posterior visualización en otros sistemas ajenos a la aplicación móvil.

- Ampliación de la aplicación con todas las titulaciones que oferta la universidad de Málaga. Un paso intermedio sería poder usar la aplicación al menos en las titulaciones de la ETSIT.
- Desarrollo de la aplicación móvil en otros sistemas operativos para dispositivos móviles.

En este caso, aún existiendo multitud de sistemas operativos para dispositivos móviles en la actualidad, hay uno con la segunda cuota de mercado más alta por detrás de Android. Este es el de los dispositivos iPhone, iOS.

- Posibilidad de introducir actividades personalizadas en el horario (actividades extraescolares, cursos, seminarios), para disponer de toda la actividad docente en un mismo horario.

- Posibilidad de crear notas o tareas pendientes por parte del usuario para así poder tener toda la información relativa a las tareas del alumno en la misma aplicación.
- Integración del servidor de este proyecto, en el servidor de la Universidad de Málaga para su puesta en marcha.

3.2 Conclusiones

Mi principal objetivo a conseguir con este proyecto ha sido lograr en la medida de lo posible, crear una App que facilite la conexión del alumnado con los datos esenciales de su vida académica utilizados en su día a día, favorecer la misma conexión con el profesorado, además de visualizar claramente el horario de clases así como la posibilidad de concertar tutorías con los determinados profesores, entre otros objetivos. En otras palabras, crear una herramienta que haga más fácil el día a día de los alumnos que estudian cualquiera de las titulaciones que en la Escuela Técnica Superior de Ingeniería Informática se ofrecen.

Una parte importante de mi andadura durante estos meses, ha sido la figura de mi tutor que ha sabido encauzar mi camino y resolver las dudas que iban surgiendo.

Después de meses de duro trabajo, con este Proyecto de fin de Grado veo finalizada una etapa de mi vida, con este proyecto he aprendido más de lo que en un principio mis expectativas preveían, ha sido un trabajo intenso, donde cada día podía avanzar un paso y en ocasiones retroceder dos, pero aún así he intentado en todo momento poner todo el entusiasmo e ilusión que un trabajo de esta envergadura se merecía.

Esto se ve acentuado por el hecho de que hasta el comienzo de este proyecto, personalmente no tenía conocimientos sobre la Programación en Android, ya que ésta solo se da durante una práctica en la asignatura de “Software para Sistemas Empotrados y Dispositivos Móviles”. Si bien es cierto que el lenguaje de programación es Java, y el mismo se da durante toda la carrera, hay muchísimo más por conocer que no se da en absoluto en la universidad. Desde cómo se consumen servicios REST en Android, a el manejo de las actividades durante todo su ciclo de vida. Y eso solo por mencionar unos detalles.

Por todo ello ha supuesto un gran reto personal y profesional cuanto menos, por todo lo que he ido aprendiendo a lo largo de este tiempo, poniéndome a prueba constantemente obligándome a averiguar la manera de saltar cada obstáculo que me impedía continuar.

Por primera vez desde el primer año de carrera, uno tiene por delante un proyecto que no es trivial, o que al menos no está preparado de antemano para su resolución por parte del profesor como pueden ser las prácticas que durante los cursos hemos realizado en las distintas asignaturas. Por primera vez estás solo (o casi), y si no sabes cómo funciona cualquier aspecto del proyecto, eres tú y solamente tú quien tiene que aprenderlo, dominarlo y llevarlo a la práctica. Esto lo considero muy útil en la futura vida profesional del alumno pues en el mundo laboral, no se tiene un profesor que esté dispuesto a solucionararte un problema, o explicarte lo que no entiendas de una determinada tecnología.

Aunque se hace duro mientras se lleva a cabo, una vez que se termina, la sensación de orgullo y satisfacción del trabajo realizado es inigualable. Una experiencia necesaria después de tantos años de estudio.

Este proyecto ha supuesto un aprendizaje mucho más allá de los meros conocimientos técnicos aprendidos de manera autodidacta. Ha supuesto un crecimiento personal inestimable. Una adquisición del conocimiento que se tiene sobre el trabajo personal de uno mismo que de ninguna otra manera se adquiere.

He podido lograr que una idea que vagaba por mi mente, se convirtiera en una realidad, algo que resulta increíblemente gratificante. De hecho, es uno de los aspectos que personalmente más valoro dentro del ámbito profesional. Nuestra profesión nos permite crear. Fabricar algo donde antes no había nada. Algo con utilidad, con un fin determinado, un efecto y un resultado en el mundo real.

En definitiva, ha sido una experiencia dura, estresante, solitaria, pero que recomendaría a todos y cada uno de los estudiantes. Por lo que estoy muy a favor de que en la actualidad el sistema de los grados ya implantado, sea obligatorio para todas las titulaciones sean de la rama que sean.

Además de todo ello, está el hecho de que para desarrollar un proyecto hay muchísimas formas de llevarlo a cabo. En esto, la programación está considerada por muchos como una especie de arte, donde hay infinidad de caminos que seguir para un determinado fin pero solo, la inspiración, la experiencia personal y el gusto de cada desarrollador definirá dicho camino y en consecuencia poniendo un poquito de él en el resultado final.

“Cuando el hombre cesa de crear, deja de existir” Lord Byron

REFERENCIAS BIBLIOGRÁFICAS

- Recolección de requisitos

http://www.ecured.cu/index.php/T%C3%A9cnicas_de_recopilaci%C3%B3n_de_requisitos

- Ingeniería de requisitos

http://sedici.unlp.edu.ar/bitstream/handle/10915/4057/2_-_Ingenier%C3%ADa_de_requerimientos.pdf?sequence=4

- Documentación de Android

<http://developer.android.com/reference/org/w3c/dom/Document.html>

APÉNDICE A. TÉRMINOS Y DEFINICIONES USADOS EN LA MEMORIA

- DUMA: Directorio de la UMA.
- UMA: Universidad de Málaga.
- GlassFish: Servidor de aplicaciones de software libre desarrollado por Sun Microsystems, compañía adquirida por Oracle Corporation, que implementa

tecnologías definidas en la plataforma Java EE y permite ejecutar aplicaciones que siguen esta especificación.

- ETSII: Escuela Técnica Superior de Ingeniería Informática.
- MySQL: Sistema de gestión de bases de datos relacional multihilo y multiusuario.
- Mockups: Modelo a escala o tamaño real de un diseño, utilizado para la demostración, evaluación, promoción y otros fines.
- TFG: Trabajo de fin de grado.
- IDE: Ambiente de desarrollo integrado.
- NetBeans: Entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java.
- UpperCamelCase: Estilo de escritura que se aplica a frase o palabras compuestas. En él la primera letra de cada una de las palabras es mayúscula.
- Hilo de ejecución: La unidad de procesamiento más pequeña que puede ser planificada por un sistema operativo.
- BD: Base de datos.
- Login: El proceso mediante el cual se controla el acceso individual a un sistema informático mediante la identificación del usuario utilizando credenciales provistas por el usuario.
- Logout: El proceso mediante se cierra el acceso personal a un sistema informático, al cual anteriormente se había realizado el login.
- Multihilo: Las unidades centrales de procesamiento con capacidad multihilo tienen soporte en hardware para ejecutar eficientemente múltiples hilos de ejecución.
- Java: Lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible.
- Google Calendar: Calendario electrónico desarrollado por Google.
- Gmail: Servicio de correo electrónico proporcionado por Google.

- Clave primaria: Campo o combinación de campos que identifica de manera única a cada fila de una tabla.
- ETSIT: Escuela Técnica Superior de Ingeniería de Telecomunicaciones.